

Coordinated Science Campaign Scheduling for Sensor Webs

Robert Morris¹
Jennifer Dungan¹
Will Edgington²

1. NASA Ames Research Center
Email: morris@email.arc.nasa.gov
2. QSS Group Inc.

Jenny Williams³
Jean Carlson³
Christopher Bruno³
and Terri Wood⁴

3. Computer Sciences Corporation
4. NASA Goddard Space Flight Center

Abstract—In the past decade, the number of Earth observation satellites has burgeoned. Present and future Earth observing missions will continue to study different aspects and interacting pieces of Earth’s hydrosphere, lithosphere, atmosphere and biosphere. Scientists are designing increasingly complex, interdisciplinary campaigns to exploit the diverse capabilities of multiple Earth sensing assets. Currently, the scheduling of scientific observations for satellites in low Earth orbit is conducted independently by every mission operations center. There is a lack of information infrastructure to enable the scheduling of coordinated observations involving multiple sensors. This paper proposes a software architecture and describes a prototype system called DESOPS (Distributed Earth Science Observation Planning and Scheduling) to address this deficiency.

I. INTRODUCTION

NASA’s Earth Science vision emphasizes the importance of establishing a tighter link among Earth Science models, data analysis, and observational activities at all relevant spatial and temporal scales. To enable such a tight linkage, there needs to be an associated information infrastructure binding the cycle of observation, on-board data handling and computing, transmission to ground, storage, data mining and product distribution to support activities such as inverse modeling, data assimilation and model evaluation. Furthermore, potential future remote sensing environments may include large numbers of networked sensors that are frequency-agile and capable of multi-scene observations from different space vantage points.

This paper provides an overview of a system that addresses the need for capabilities related to the coordination of observations. The system is based on a methodology called model-based observing. *Model-based observing* is the process of allocating and scheduling sensing resources based on the goal of validating a specific hypothesis derived from an Earth science model. Model-based observing allows observation scheduling to be *campaign-driven*, where a campaign is defined as a systematic set of activities undertaken to meet a particular science objective. Campaign goals require the collection of data on several variables on different observing resources at different times and locations.

In the next section we present the overall architecture for model-based observing that links the Earth Science community to observation resources. Part of the architecture forms the

set of capabilities for coordinating observations, which is the focus of the remainder of the paper. These capabilities are organized into a set of components of a system, called DESOPS (Distributed Earth Science Operations Planning and Scheduling System).

II. ARCHITECTURE FOR MODEL-BASED OBSERVING

Model-based observing requires coordinating the assignments of observation tasks among a collection of remote sensors or sub-orbital platforms such as ground-, airborne-, and balloon sensors, configured into an organization of some sort (e.g. a train or a sensor web) [4]. We assume a separate mission operations center for managing the daily activities of each sensor. Consequently, the system for coordinating observations provides an added layer between the individual science community and mission operations planning. The coordination layer allows a user to create a *campaign plan* that is then executed by submitting individual observation requests to one or more of a set of relevant missions. Missions have the option of rejecting the request, which automatically triggers re-submissions of new requests or campaign replanning. The overall architecture is displayed in Figure 1. The coordination layer is labeled DESOPS (Distributed Earth Science Observation Planning and Scheduling). DESOPS consists of the information infrastructure for constructing campaign plans involving a collection of sensors, and enables more direct contact between Earth scientists and the mission planning process. The next sections describe these capabilities in more detail. A more formal description of the computational problem and solution algorithms used by DESOPS can be found in [8].

III. DESOPS CAPABILITIES

DESOPS is a multi-user coordinated scheduler of multiple sensors. Its core function is to generate and execute Earth science campaign plans. Campaign plan generation includes managing a set of user-specified requirements that provide constraints on feasible plans, employing a set of optimization criteria for ordering feasible plans based on user preferences and utilizing models of the missions and sensors. Plan execution consists of formatting and submitting requests to missions,

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 25 JUL 2005		2. REPORT TYPE N/A		3. DATES COVERED -	
4. TITLE AND SUBTITLE Coordinated Science Campaign Scheduling for Sensor Webs				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) NASA Ames Research Center				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release, distribution unlimited					
13. SUPPLEMENTARY NOTES See also ADM001850, 2005 IEEE International Geoscience and Remote Sensing Symposium Proceedings (25th) (IGARSS 2005) Held in Seoul, Korea on 25-29 July 2005. , The original document contains color images.					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 4	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

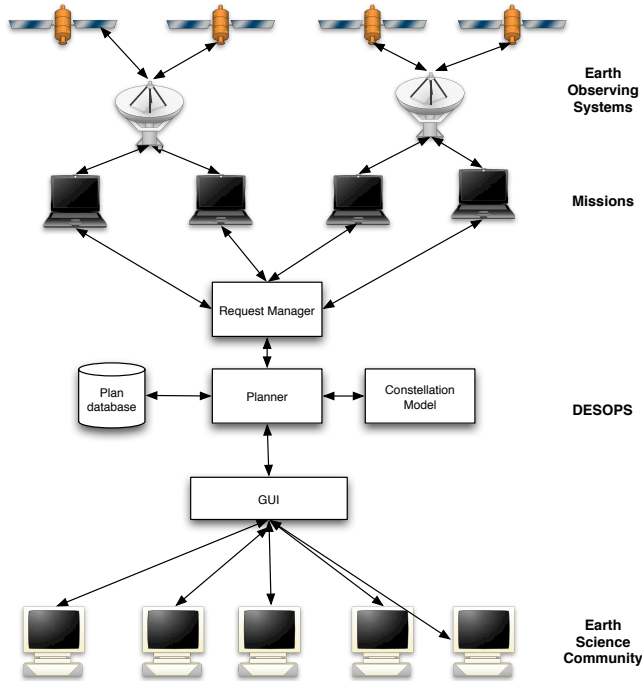


Fig. 1. A general architecture for model-based observing

continuous monitoring and, if necessary, replanning based on the results of submitting requests and other unexpected events.

A. Constellation Model and User Inputs

A **constellation model** consists of five parts: a definition of a set of *sensors* each associated with a cost for using it to take an observation; second, a *time domain*, a finite set of totally ordered values naturally interpreted as the set of days in which some observation can be taken or some other event of interest happens; third, a *geographic domain* for identifying the locations and extents of regions to be measured (for example, a region of interest could be specified as a set of latitudes and longitudes to define arbitrary polygons on the Earth); fourth, *satellite orbit* model for determining the set of sensor viewing times for a specified region of interest; and fifth, for each sensor a *mission model* that describes constraints on the process by which tasks on the sensor are scheduled by the mission that manages it.

Users provide inputs through a **graphical user interface**. User inputs consist of a set of *measurements* that make up the campaign, a set of *exogenous events*, and a set of *constraints*. If desired, the user can specify a partial order on the measurements, indicating relative importance of each in fulfilling the goals of the campaign. Exogenous events (like a fire) are needed because campaigns are often planned around them, and it is necessary to be able to specify temporal or spatial relationships between these events and observation activities. The set of user-specified constraints on a campaign restrict the way the campaign can be executed. DESOPS supports five kinds of constraints:

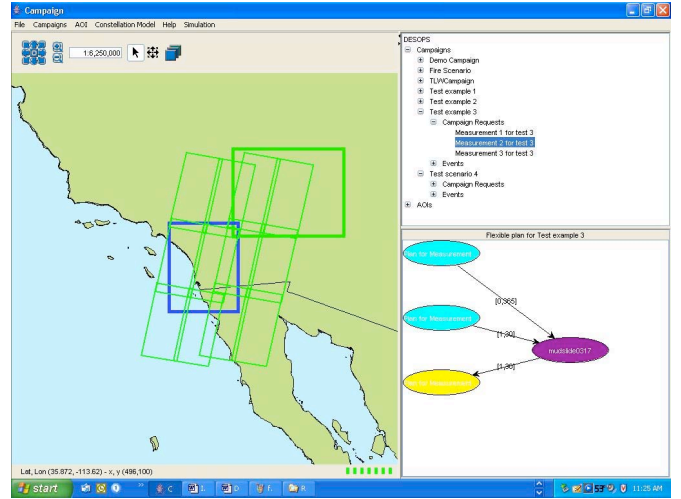


Fig. 2. Graphical user interface for defining coordinated campaigns. The blue box represents the region of interest geographic constraint, green boxes represent view paths that satisfy that constraint.

- 1) *sensor constraints* that define a list of sensors through which a measurement to be acquired, with optionally defined preferences for sensors on the list;
- 2) *temporal constraints*, either in the form of a *time window* (a range of times) for taking a measurement, or ordering restrictions, either between pairs of measurements, or between measurements and exogenous events. In addition, the user may optimally specify *preferences* for time values for these constraints; for example, a user may express a preference for measurements to be "as close as possible" to others, following the approach taken in [3]. The user may also specify *beliefs* about when events are expected to occur, following the approach in [2], where these beliefs are expressed quantitatively in terms of probability distributions.
- 3) a *geographic constraints* for each measurement, each specified as a set of latitudes/longitudes;
- 4) a *constraints on data characteristics* for specifying requirements for cloud-free observations, for example, and
- 5) a *cost constraints*.

The main screen of the interface is displayed in Figure 2. This screen shows a map for specifying regions of interest for a campaign, a flexible plan (defined in more detail below) and a textual representation of a campaign as a hierarchy of measurements and constraints. The overpass swaths for one of the requested satellites has also been computed automatically and is visually displayed.

DESOPS must be able to evaluate "good" schedules. Examples of evaluation criteria include 1) *world-feasibility*; a solution is world-feasible if it satisfies all constraints and is optimal with respect to the expected behavior of the exogenous events; 2) *minimal cost* where the cost of a solution is the sum of the cost of The sensors used on each measurement; 3) *temporally preferred*; solutions that maximize satisfaction of time constraints; and 4) *resource preferred*; solutions that

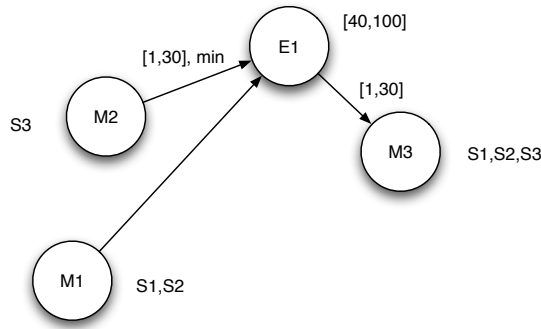


Fig. 3. A simple Flexible Plan. Each node of the network represents a measurement or event. Directed arcs depict temporal orderings, labeled with the duration between event(s) and measurements(s). The sensors are listed with each measurement.

maximize the overall satisfaction of sensor preferences. The best assignments will be those that satisfy a weighted combination of these criteria.

B. Planning Campaigns

A *flexible plan* is a concise representation of a set of possible solutions to a campaign scheduling problem. The role of the **Planner** is to build and manage flexible plans. First, the planner constructs an initial flexible plan based on user inputs. Second, new constraints can be added by propagating the effects of an initial set of temporal orderings. In particular, the planner generates start times for each sensor in the domain of each measurement from *view paths* over specified regions of interest during specified time windows. A view path is the intersection of a specified region of interest with the path followed by a satellite over the user-specified time window. In DESOPS, view paths are generated by conducting a web search for this data from mission web sites. Alternatively, it is possible to generate this data directly through the use of simulators such as STK (Satellite Tool Kit). Converging on a flexible plan is an iterative process in which the user is allowed to view and revise the inputs to the problem.

A flexible plan can be represented as a network of nodes representing events or measurements, and directed arcs labeled by constraint information. An example is found in Figure 3. The plan consists of three measurements and one event. The constraint $[40, 100]$ represents the belief that event $E1$ is expected to happen sometime between day 40 and day 100 of the campaign. The other constraints represent temporal ordering constraints; for example, the label between $M1$ and $E1$ expresses the constraint that $M1$ should occur between 1 and 30 days before $E1$, with a preference for times as close to $E1$ as possible. The sensor constraints are also attached to each measurement in the plan.

C. Plan Execution

An *observation request* for a measurement is an assignment of a sensor, a time, and a location to the measurement. A *feasible observation schedule* is a sequence of observation

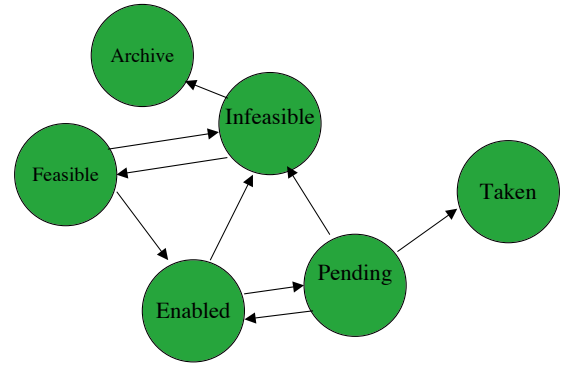


Fig. 4. A state transition model for measurements. States and possible transitions between them are depicted.

requests that satisfy the user specified constraints. In general, a flexible plan gives rise to a number of feasible observation schedules. User specified preferences induce these orderings. The **Request Manager** incrementally executes a feasible observation schedule by submitting observation requests to missions. The Request Manager also *monitors* the state of the executing plan, and initiates *rescheduling* activities where necessary. To carry out these functions the Request Manager implements an execution strategy for dealing with uncertainty in the execution environment and manages a state-transition model for monitoring the progress of the plan.

An *execution strategy* is based on a mission model and information about exogenous events. The mission model advises the Request Manager on matters related to which mission is most likely to be able to fulfill a request, as well as how and when to submit the request. For example, the mission model will contain a *load profile* for each sensor, which indicates the percentage of time the sensor has been idle during a specified period. The Request Manager applies this information by preferring sensors with a smaller load. Second, a mission model contains formatting rules for request submission. Third, a mission model contains requirements for when to submit requests such as deadlines for submitting requests based on the mission-scheduling process.

A *state-transition model* identifies possible states of the overall plan, the component measurements in the plan, and, for each measurement, the state of each associated observation request. The model also defines transitions between states. The Request Manager implements the plan state-transition model as the mechanism for monitoring the progress of the plan. The Request Manager observes whether enabling conditions for a transition are met, and, if they are, records the change in state. The state-transition model also allows the Request Manager to detect when a campaign has failed during execution, which triggers a suspension of the campaign and notification to the user for rescheduling purposes.

Figure 4 shows a state transition model for a measurement. A measurement starts in a feasible state. It becomes enabled when the temporal preconditions for taking the measurement

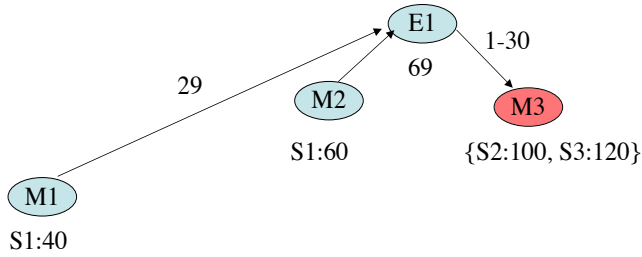


Fig. 5. A replanning scenario. The occurrence of $E1$ at time 69 has made it impossible to schedule an observation of $M3$ that satisfies the constraint between $E1$ and $M3$. The user must decide whether to relax the constraints on the plan to restore its feasibility.

are met (for example, an exogenous event happens or a dependent measurement has been acquired). It becomes infeasible if the constraints make it impossible for it to be taken; this can happen, for example, if all submissions of requests for the measurement are rejected. Otherwise, a measurement is pending if at least one request for the measurement has been submitted. If a mission accepts the request and the image is acquired, the measurement enters the terminal node *Taken*. The user may decide during execution to use data in an archive to acquire the needed data. If so, the Request Manager no longer submits requests for the observation to the missions.

D. Replanning

As the campaign plan executes, observations or exogenous events happen, which can potentially render a campaign plan infeasible.

At this point, the user decides whether to restore feasibility to the plan or to abort it. Plans are made feasible by relaxing constraints that contributed to making the plan infeasible. Figure 5 shows a simple plan that was made infeasible during execution. Exogenous event $E1$ happened at time 69. A constraint requires measurement $M3$, which has yet to occur, happen between 1 and 30 days after $E1$. $M3$ has two observation opportunities: with sensor $S2$ at time 100, or with sensor $S3$ at time 120. Clearly, both exceed the upper bound on the temporal ordering constraint, and so this constraint is violated. The user may relax the upper bound of the temporal constraint to make the observation opportunities consistent with the plan. Alternatively, the user may add additional sensors for $M3$ that include opportunities consistent with the ordering constraint, or may decide to acquire $M3$ data through an archive.

DESOPS provides the user continuous plan execution status when requested. It also provides notification of the need for plan repair when the plan becomes infeasible during execution. Visual and textual information will be provided by DESOPS' explanation facility, using a model to map plan state information into useful textual or visual advice.

IV. IMPLEMENTATION AND ENHANCEMENTS

The DESOPS system design described in this paper is being implemented in C++ and Java. The implementation is built

upon previous work on the AMPS/MOPSS system and the EUROPA constraint-based planning system [7]. An end-to-end prototype with a subset of the capabilities described in this paper is currently being tested and evaluated.

As noted at the outset, DESOPS is one part of a broader system for realizing NASA's Earth Science vision integrating observing, analysis and modeling [1]. There are three broad classes of capabilities that offer the means of expanding DESOPS into a complete set of capabilities for accomplishing this vision. First, an integration of Earth Science domain models into DESOPS would enable the system to advise a user in formulating campaigns. For example, such models could advise users on the selection of promising regions-of-interest for developing a fire campaign. Second, the integration observation scheduling with planning for data analysis as discussed in [6] would lead to an end-to-end system for generating data products. Third, providing the automated means of transforming the results of image analysis into goals for future observation scheduling, as demonstrated on EO-1 [5] would "complete the loop" in automated campaign execution.

V. CONCLUSION

This paper has describe a set of capabilities for building and executing sequences of observations for accomplishing complex campaign goals. Observation requests generated from user inputs describing campaign goals and constraints are submitted electronically to mission operations planners, who then decide whether and how to incorporate the request into future mission schedules. The system also supports dynamic replanning in response to request rejection or unexpected changes in the observing environment. The overall approach to distributed planning has the advantage of allowing missions to maintain ultimate control over their instruments while at the same time allowing Earth scientists more visibility into the resources available for accomplishing their science objectives.

REFERENCES

- [1] NASA Earth Science Enterprise Strategy, 2003.
- [2] R. Morris et. al., *Temporal Planning with Preferences and Probabilities*, Proceedings of the ICAPPS Workshop on Constraint Programming for Planning and Scheduling, 2005.
- [3] P. Morris et. al., *Strategies for Global Optimization of Temporal Preferences*, Proceedings of CP 2004: 408-422.
- [4] B. Clement and A. Barrett, *Coordination Challenges for Autonomous Spacecraft*, AAMAS-02 Workshop Notes on Toward an Application Science: MAS Problem Spaces and Their Implications to Achieving Globally Coherent Behavior, 2002.
- [5] S. Chien et. al., *Autonomous Science on the EO-1 Mission*, International Symposium on Artificial Intelligence, Robotics, and Automation in Space (i-SAIRAS 2003), Nara, Japan, May 2003.
- [6] K. Golden et. al., *Automating the processing of Earth observation data*, Proceedings of the 7th International Symposium on Artificial Intelligence, Robotics and Automation for Space (i-SAIRAS), 2003.
- [7] D. Smith, J. Frank, and A. Jonsson, *Bridging the Gap Between Planning and Scheduling*, Knowledge Engineering Review, 15:1, 2000.
- [8] R. Morris et. al., *An Architecture and System for Scheduling Coordinated Earth Science Measurements*, Proceedings of the 8th International Symposium on Artificial Intelligence, Robotics and Automation, Munich Germany, 2005.